ACT0

Evidencia 1.-Responder en tarjetas las siguientes preguntas

1. ¿Qué es un SVC (Sistema de Control de Versiones)?

R= Son herramientas de software que ayudan a los equipos de software a gestionar los cambios en el código fuente a lo largo del tiempo.

1. ¿Qué es git?

R= **Git** es un sistema de control de versiones que permite a los desarrolladores descargar un software, realizar cambios y subir la versión que han modificado

1. ¿Qué sistemas Git gratis existen?

R= GitHub, GitLab,

1. ¿Para qué sirve el comando git?

R= Muestra solo la versión que tenemos instalada, da documentación,

1. ¿Para qué sirve el comando git init?

R= creará un nuevo repositorio local

1. ¿Para qué sirve el comando git add?

R= se usa para agregar archivos al área de preparación

1. ¿Para qué sirve el comando git commit?

R= creará una instantánea de los cambios y la guardará en el directorio git

1. ¿Para qué sirve el comando git push?

R= Se usa para enviar confirmaciones locales a la rama maestra del repositorio remoto

1. ¿Cuáles son los pasos a seguir para subir un proyecto a un repositorio?

R= Abre la terminal en la carpeta donde se alojan los ficheros fuente y escribe el siguiente código para iniciar git en dicha carpeta:

git init

git add

Otra forma de añadir un único fichero es introduciendo el comando siguiente:

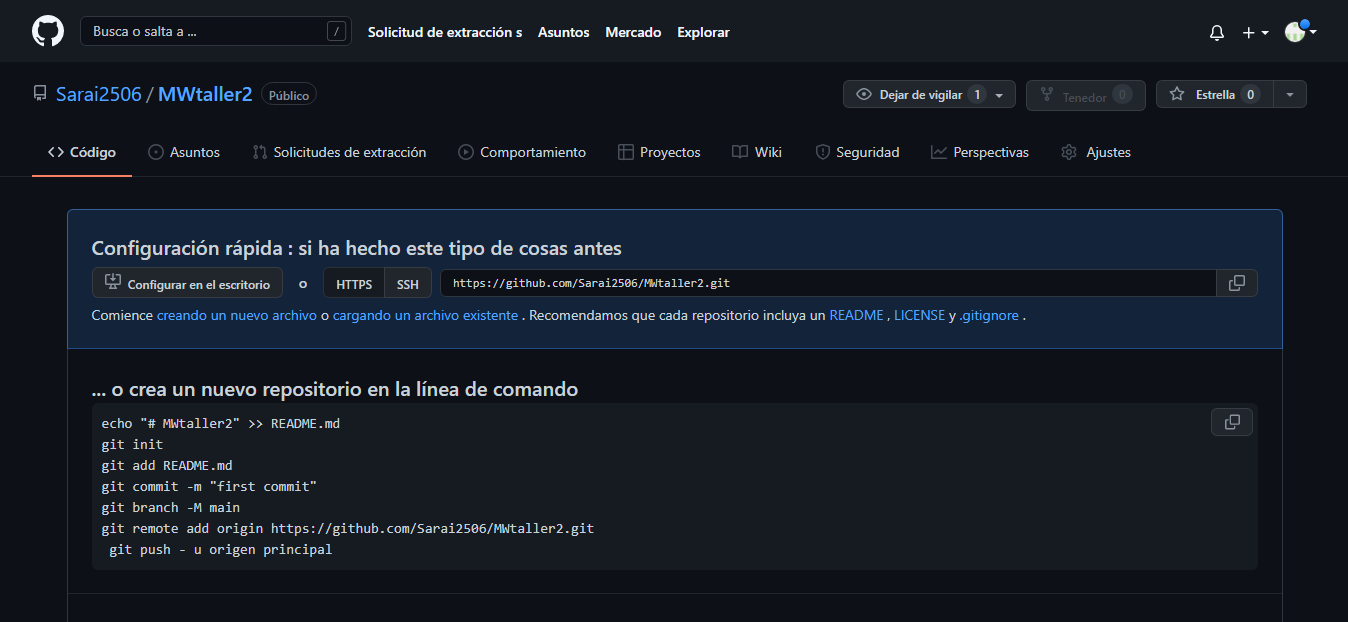
 git add nombrdelfichero

¿Por qué git te permite trabajar colaborativamente en un proyecto?

R= Porque al subir tu código al repositorio principal, cualquier compañero que trabaje en el mismo proyecto podrá actualizar su copia local y tener esos archivos.

1. ¿Qué errores o conflictos pueden surgir al trabajar colaborativamente en un proyecto con git?

R= En un proyecto de varias personas, puede pasar que cada uno tenga su rama de trabajo en su repositorio. Esto puede complicar las cosas si se quiere probar la compatibilidad de cambios de diferentes personas sin pasar por el repositorio principal.

Evidencia 2.-Capturas de pantalla de sitio en Github

ACT 1

Evidencia 3.- Responder preguntas en tarjetas

1. ¿Qué es un lenguaje de programación?

R= Es un lenguaje de computadora que los programadores utilizan para comunicarse y para desarrollar programas de software, aplicaciones, etc.

1. ¿Qué es JavaScript y para qué sirve?

R= JavaScript es un lenguaje de programación que sirve para añadir características interactivas al sitio web, que ocurren cuando los botones son presionados o los datos son introducidos en los formularios, etc.

1. ¿Qué es la interactividad en las páginas web y por qué es importante?

R= La interactividad se configura como un elemento básico, determinando la eficacia de la página Web, y facilitando a la organización y al usuario el acceso y suministro de la información.

1. ¿Con qué lenguajes se puede crear una página interactiva?

R= JavaScript, Css, HTML.

1. ¿Qué es el DOM?

R= El modelo de objeto de documento (DOM) es una interfaz de programación para los documentos HTML y XML

1. ¿Qué son los scripts?

R= Secuencia de comandos​ que se usa para designar a un programa.

1. ¿Cómo introducir JavaScript en mí mismo archivo HTML?

R= Con un <script></script>

1. ¿Cómo introducir JavaScript ligando un archivo externo a mi HTML?

R= Con un <script></script> y el atributo src

1. ¿Cómo mostrar mensajes de alerta en mi página web?

R= **alert**("Texto a mostrar");

1. ¿Cómo mostrar mensajes de confirmación en mi página web?

R= [prompt](https://www.anerbarrena.com/javascript-prompt-js-5509/): Determina si el usuario ha clickado en ‘Aceptar’, ‘Cancelar’ o si se ha introducido texto en dicha ventana modal.

1. ¿Cómo solicitarles a las personas visitantes de mi página web que ingresen alguna información?

R= Mediante de un formulario <form></form>, etc.

1. ¿Qué son las funciones en JavaScript?

R=Es un conjunto de instrucciones que realiza una tarea o calcula un valor este debe tener una entrada y devolver una salida.

Evidencia 4.- Captura de pantalla de sitio MWT2A1\_ [NombreDeLaPágina], con estructura archivos HTML

Carpetas css y js

* La página mínimamente deberá tener las etiquetas [ título, subtítulo, encabezado, pie de página, imágenes, ligas, párrafos y menú]
* En css deberan tener minimanente las propiedades font-family, font-size, color, textalign, width, height, border-style, border-width, border-color, margin, padding y background-image
* Archivo de extensión .js con las funciones alert, confirm y prompt. Subir todo lo nuevo a la cuenta git

ACT2

Evidencia 5.- Responder preguntas en tarjetas

1. ¿Qué es la consola del navegador y para qué sirve?
2. ¿Cómo mostrar un mensaje de texto en la consola del navegador?
3. ¿Cómo mostrar un mensaje de error?
4. ¿Cómo mostrar tablas?
5. ¿Cómo limpiar la consola?
6. ¿Cómo mostrar un temporizador en la consola?
7. ¿Cómo realizar comentarios en código JavaScript?
8. ¿Qué es una variable en JavaScript y cuál es su utilidad?
9. ¿Qué tipos de variables existen?
10. ¿Cuál es el alcance o ámbito de las variables?
11. ¿En qué consiste la nomenclatura CamelCase y en dónde se utiliza?
12. ¿Qué es un tipo de dato y para que se utilizan?
13. ¿Qué tipos de datos existen y para qué es cada uno?
14. ¿Qué tipos de operadores lógicos existen y cuál es la utilidad de cada uno?
15. ¿Cómo se maneja una fecha en JavaScript?
16. ¿Qué función de JavaScript muestra la fecha actual del sistema?

Elaborar:

Tres cuadros sinópticos:

* el primero sobre el tema la consola del navegador (preguntas 1, 2, 3, 4, 5 y 6).
* el segundo acerca de variables (preguntas 8, 9 y 10).
* el tercero sobre los tipos de datos (preguntas 12 y 13).

Se supervisará que dichos cuadros sinópticos: mantengan una estructura que vaya de lo general a lo particular; y que estén conformados por conceptos primarios (por ejemplo variable), secundarios (por ejemplo tipos de variables), terciarios -de ser el caso- (por ejemplo string), definiciones, explicaciones y/o funciones de los conceptos.

Fichas de consulta (con el formato hasta ahora trabajado) correspondientes a las preguntas 7, 11, 14,15 y 16.

Evidencia 6.- Ejercicios:

1. Script que imprima en pantalla un “Hola mundo”; la cadena debe ser declarada como una variable.

2. Script que realice la suma de dos números y muestre el resultado en pantalla.

3. Script que muestre un mensaje únicamente en la consola del navegador.

4. Script que determine si un número es par

ACT3

Evidencia 7.- Solución de anexo MWT2A3\_Anexos.pdf

Evidencia 8.-Ejercicio El ingreso del precio de un artículo y la cantidad de artículos que lleva el cliente. Se debe ingresar por teclado:

1. El número de artículos (deben de ser números enteros).

2. El precio de cada artículo. Se debe mostrar lo que debe de pagar el cliente.

ACT 4

Evidencia 9.- Captura de pantallas de 3 páginas con formularios y encerrarlos con una herramienta de imagen.

Evidencia 10.- Crear página con formulario de tienda Línea MínimamentePrincipio del formulario y 3 input Agregarle css al formulario, en la estructura definida js, css y HTML.

ACT5

Evidencia 11.- Resolviendo preguntas de arreglos:

1. ¿Qué son los arrays o arreglos en JavaScript?
2. ¿Cuál es la sintaxis de un arreglo?
3. ¿Cómo se manejan las posiciones de los elementos en un arreglo?
4. ¿Cómo crear un arreglo?
5. ¿Qué tipos de datos puede almacenar un arreglo?
6. ¿Qué es un método de un arreglo?
7. ¿Para qué me sirven los arreglos?
8. ¿Para qué me sirven los métodos de estos?
9. ¿Los datos almacenados en un arreglo son temporales o permanentes?

Evidencia 12.- Ejercicio:

* Crear un arreglo y almacenar cada producto de su lista.
* Calcular el tamaño del arreglo. (length) Meter un elemento al final. (push)
* Meter un elemento al inicio. (unshift) Sacar un elemento del final. (pop)
* Sacar un elemento del inicio. (shift) Ordenarlo. (sort)
* Ponerlo en reversa. (reverse)
* Copiar una porción del arreglo. (slice)
* Agregar, reemplazar y eliminar elementos. (splice)
* Crear un nuevo arreglo y concatenar ambos. (concat).
* Subir a repositorio del taller la carpeta MWT2A5\_Arreglos con el ejercicios
* Subir a carpeta “MWT2A5\_FichasTrabajo” las preguntas resueltas

Evidencia 13.- Resolviendo preguntas:

1. ¿Qué son los iteradores o ciclos en JavaScript?
2. ¿Con qué otro nombre aparte de los mencionados se le conocen?
3. ¿Por qué crees que les llaman así?
4. ¿Cuál es la sintaxis de un ciclo for, de un forEach, de un while y de un do-while?
5. ¿Qué diferencias hay entre cada uno de ellos? ¿Para qué sirven?
6. Subir a repositorio

Evidencia 14.- Ejercicios:

* Crea carpeta "MWT2A5\_Iteradores"
* Escribir un script que mediante forEach, for, while y do-while muestre el arreglo elemento por elemento.
* Utilizando while:
  + Mostrar los números del 1 al 10 de forma ascendente.
  + Mostrar los números del 1 al 10 de forma descendente.
* Utilizando for:
  + Mostrar la tabla de multiplicar de un número.
  + Mostrar todas las tablas de multiplicar
* Utilizando do-while:
  + Pedir 10 datos de precios e irlos sumando, al final debe mostrar el total de la suma.
  + Pedir cinco datos de precios y mostrar el promedio de estos.
* Utilizando el ciclo que la participante crea que se adapte mejor, escribir un script que recorra un arreglo de números y encuentre el número máximo y el número mínimo del arreglo.
* Utilizar condicionales.

ACT 6

Evidencia 15: Resolviendo anexo MWT2A6\_Anexo

Evidencia 16: Formulario para agregar nuevos productos

ACT 7

Evidencia 17: Investigación funciones (valores, alcance y llamadas)

Evidencia 18: Resolver MWT2A7\_Anexos.

Evidencia 19:

1. Utilizar ‘El método eval () ‘para mostrar el nombre completo de un producto como una sola cadena de caracteres.
2. Además se deberá utilizar “La función isNaN” para verificar que todos los precios de los productos sean números enteros.
3. También se utilizará la función “show\_props () “para llamar a la lista de productos del catálogo y mostrar los productos que haya en existencia.
4. Por último, se tendrá que definir una función para listar los productos por orden alfabético.

Evidencia 20: Realizar los siguientes ejercicios en funciones:

• Implementar en el código una función recursiva que realice la suma de los productos que se han agregado al catálogo.

• ¿Qué otras funciones de los códigos de la actividad 6 podrían ser útiles en el manejo del catálogo?

• Ejecutar las funciones que consideres pertinentes y anotar las mejoras realizadas en el código así como de qué manera estas le ayudan a manejar sus elementos

ACT 8

Evidencia 21: Investigando:

1. ¿Qué es el DOM?
2. ¿Para qué sirve? y ¿cómo funciona?
3. ¿Qué es el árbol DOM?
4. ¿Qué es un nodo, en el DOM? y ¿Cómo acceder a los nodos del DOM?

|  |  |
| --- | --- |
| Cómo | Métodos |
| Encontrar elementos por ID | document.getElementById(id) |
| Encontrar elementos por nombre de etiqueta... |  |

1. Cómo realizar acciones directas dentro del DOM:

* Encontrar elementos HTML por id, clase y por nombre de etiqueta. (getElementById(id), getElementsByTagName(nombre), getElementsByClassName(nombre))
* Cambiar el valor de un atributo de un elemento. (element.attribute = new value)
* Cambiar el estilo de un elemento HTML. (element.style.property = new style)
* Encontrar elementos HTML mediante selectores CSS (querySelectorAll())
* Cambiar el HTML interno de un elemento (element.innerHTML = “nuevo contenido”)
* Cambiar el flujo de salida HTML. (document.write(texto))

2. Cómo realizar acciones con los nodos del DOM: ○ Crear (document.createElement (elemento))

* Agregar(document.appendChild(elemento)) ○ Remover(document.removeChild(elemento))
* Reemplazar(document.replaceChild(nuevoElemento, antiguoElemento)

3. ¿Qué diferencias existen entre document.write e innerHTML? ¿En qué casos se debe utilizar cada uno?

Template Strings e introducción a eventos.

Realizar un organizador gráfico en forma de mapa conceptual respondiendo las siguientes preguntas:

1. ¿Qué son los template strings?
2. ¿Cómo utilizarlos? y ¿Para qué sirven?
3. ¿Qué es un evento?
4. ¿Cómo puedo utilizar un evento para cambiar un comportamiento en mi página?
5. ¿Cómo agregar un evento a un botón?

4. Mi árbol de nodos.

Elaborar un árbol de jerarquía de nodos representando el código HTML de la página principal del proyecto TiendaLinea[NombreDelProyecto], podrán utilizar cualquier herramienta para dibujarlo. SUBIR en la carpeta “MWT2A8\_OrganizadorGráfico”.

ACT 9

Evidencia 22: Realizar tabla

|  |  |  |  |
| --- | --- | --- | --- |
| Evento | Descripción | Elementos para los que está definido | Sintaxis |
| onfocus | Un elemento obtiene el foco | <button>, <input>, <label>, <select>, <textarea>, <body> | <element onfocus=”myscript”> |

Mínimamente estos elementos:

* onclick
* ondblclick
* onmousedown
* onmousemove
* onmouseover
* onmouseout
* onkeydown
* onkeypress
* onkeyup
* onfocus
* onblur
* onchange
* onreset
* onresize
* onselect
* onsubmit
* onunload

Debajo del cuadro incluir los siguientes métodos, junto con la sintaxis y explicación de cada uno:

* addEventListener()
* removeEventListener()

SUBIR en la carpeta MWT2A9\_Eventos

Evidencia 23: Realizar la página pedido.html

1. El registro de los pedidos se agregará de forma dinámica por lo que ahora el formulario constará de solo tres campos: nombre, precio y cantidad.

2. Los cálculos del total y subtotal se calcularán dinámicamente.

3. Implementar los eventos correspondientes para el funcionamiento del registro de pedidos y para mostrar el resumen de estos.

4. En el resumen se deberá visualizar el listado de los productos pedidos, el total y subtotal.

5. La página web deberá presentar estilo.

6. Al final de la actividad, la página web tendrá que visualizarse sin ningún error.

SUBIR a repositorio.

ACT 10

Evidencia 24: Investigando principales características de JavaScript, node.js, Express.js y middleware

● JavaScript o Lenguaje de programación o Interpretado o Tipado débil

● Node JS o Entorno de ejecución para JavaScript (ejecutar JavaScript fuera de un navegador web) o Asíncrono y orientado a eventos o Contiene un gestor de paquetes npm

● Express JS o Paquete/ Librería de npm o Framework para realizar páginas web

● Middleware o Ciclo request - response o Se ejecuta una función una después de otra (en orden) o ¿De qué se compone una función middleware en express? o Tipos de middleware

Evidencia 25: Resolver MWT2A10\_Anexo

Evidencia 26: Investigar 1)

¿Qué es express application generator?

¿Para qué sirve?

2) ¿Qué es un motor de plantillas?

3) ¿Qué motores de plantillas admite express application generator?

Investiga al menos dos

4) ¿Con qué comando podemos generar un proyecto sin motor de plantillas definido?

5) Crea un proyecto con alguno de los motores de plantillas investigados en el punto 3 (proyecto 1)

6) Crea un proyecto sin motor de plantillas definido (proyecto 2)

7) Observa la diferencia entre los archivos creados en ambos proyectos y realiza una comparación

8) ¿Cómo se puede ejecutar el proyecto?

9) Ejecuta el proyecto 1. ¿Qué observas?

10) Ejecuta el proyecto 2. ¿Qué observas?

11) ¿Existe alguna diferencia entre sus resultados?

12) En caso de que existieran errores ¿qué los originó? ¿Cómo se podrían resolver?

13) Adjunta una captura de la estructura de carpetas generada e investiga brevemente qué tipo de archivos almacena cada carpeta

14) ¿En qué se diferencia a la estructura generada con la receta “Servidor a la Express”? ¿Existen similitudes?

15) ¿Por qué consideras que se necesitan más carpetas, conforme aumenta la complejidad del proyecto?

ACT 11

Evidencia 27: Investigando

* 1. Protocolo HTTP y TCP
  2. API Rest
  3. Métodos HTTP (definición, uso de GET y POST) Realizar un bosquejo de la comunicación cliente-servidor en la web (es posible retomar el esquema realizado en el taller anterior: MWT1A9) agregando los conceptos de los protocolos HTTP y TCP, destacando la estructura de un request y response en HTTP, el lugar en que se situaría el API Rest y la relación de lo anterior con Express.js.

Evidencia 28 Realizará un formulario que implemente los métodos GET y POST en Express.js.

El formulario tendrá como propósito saber quién tiene más participación en el hogar ante determinadas actividades, por lo tanto deberá tener campos donde se pueda capturar esta información, algunos ejemplos de actividades pueden ser: lavar los platos, cocinar, barrer, planchar, ir al súper, etcétera.

ACT 12

Evidencia 29: Investigando métodos HTTP faltantes (PUT y DELETE), poniendo especial atención en la acción que realiza cada uno y sus principales características

Evidencia 30: Creará una API REST que implemente la consulta, creación, actualización y eliminación en el backend del servidor del punto de venta.

ACT 13

Evidencia 31: Investigando

* ¿Qué es el MVC?
* ¿Por qué es importante el MVC?
* ¿Es posible aplicar o ya está siendo aplicado el MVC a nuestro proyecto?
* ¿Se puede aplicar del todo hasta ahora o hay alguna pieza faltante?
* ¿Quién sería el modelo? ○ ¿Quién sería la vista? ○ ¿Quién sería el controlador?
* ¿El MVC es algo exclusivo del desarrollo web?
* ¿A qué hace referencia CRUD?
* ¿Qué partes del CRUD ya han sido aplicadas al proyecto?

Evidencia 32: CREATE

○ Agregar el código correspondiente para que los productos registrados se guarden en el servidor, en forma de un arreglo en el backend. Adicional a ello se debe modificar la parte de pedidos para que el precio se tome de la información de productos en el servidor. READ ○ Generar una nueva página que despliegue el listado de productos registrados obtenidos del servidor en el frontend, con el nombre “ListadoProductos.html” UPDATE

○ Modificar la página de listado de productos para que sea posible desde ahí actualizar el precio de un producto. DELETE

○ Agregar el proceso de borrado en la página del listado de productos (backend y frontend).

ACT 14

Evidencia 33: Investigando

1. ¿Qué es una base de datos?
2. ¿Qué es MongoDB?
3. ¿Cuál es la función para crear un objeto con MongoDB y JavaScript?
4. ¿Cuál es la función para obtener un objeto con MongoDB y JavaScript?
5. ¿Cuál es la función para actualizar o modificar un objeto con MongoDB y JavaScript?
6. ¿Cuál es la función para eliminar un objeto con MongoDB y JavaScript?
7. ¿A qué pertenecen las cuatro acciones correspondientes a las cuatro preguntas anteriores?
8. ¿Cuáles son las instrucciones o pasos para configurar una base datos en MongoDB con Node.js?
9. ¿Qué empresas utilizan MongoDB en sus sitios web?

Realizar una tabla como la que se muestra a continuación, únicamente con la información correspondiente a las preguntas 3 – 6. Del resto de ellas, deberá realizar fichas de consulta como las trabajadas en las actividades anteriores. Funciones con MongoDB y JavaScript Para crear un objeto Para obtener un objeto Para actualizar o modificar un objeto Para eliminar un objeto Evidencia 34: Realizar un formulario nombre completo, número de casa, número de celular y relación Realizar otra página que muestre los registros ingresados. Realizar todo lo de la tienda guardando ahora en Mongo DB